JREEC
JOURNAL RENEWABLE ENERGY ELECTRONICS AND CONTROL
homepage URL : https://ejurnal.itats.ac.id/jreec

DETEKSI SERANGAN PING FLOOD PADA SERVER CCTV

Dani Raisman1, Refdi Andri2, dan Nelly Khairani Daulay3

Universitas Bina Insan1,2,3

INFORMASI ARTIKEL

Jurnal JREEC – Volume 03 Nomer 01, Juni 2023

Halaman:
48-58
Tanggal Terbit :
06 Juni 2023

DOI:
10.31284/j.JREEC.2023.v31i.4520

ABSTRACT

The development of information technology, especially computer networks, allows the exchange of information that is easy, fast, and increasingly complex. Computer network security must be considered in order to maintain the validity and integrity of data and information residing in the computer network. The problem that arises in the CCTV control center is that there is no security against the detection of attacks that can occur at any time, for example a ping flood attack. Ping flood itself can be interpreted as a simple denial of service attack in which the attacker floods the victim with "echo request" (ping) packets in the ICMP protocol. To overcome the problems faced by the Lubuklinggau City Police control center in carrying out CCTV server security, the solution offered is to build an Intrusion Detection Server (IDS). IDS itself can read incoming and outgoing data packets automatically which will provide a report (log) to the network administrator. One of the most widely used IDS tools is Snort. Snort has several advantages compared to other IDS software, including source code that is small in size, compatible with many operating systems, fast in detecting network attacks, easy to configure and is open source.

Kata kunci: *Network security, IDS, Snort*

EMAIL

daniraisman1995@gmail.com 1
refdia3@gmail.com 2
nellykhairanilestari@gmail.com 3

ABSTRAK

Perkembangan teknologi informasi, khususnya jaringan komputer memungkinkan terjadinya pertukaran informasi yang mudah, cepat, dan semakin kompleks. Keamanan jaringan komputer harus diperhatikan guna menjaga validitas dan integritas data serta informasi yang berada dalam jaringan komputer tersebut. Permasalahan yang timbul di pusat kendali CCTV ini adalah belum adanya pengamanan terhadap pendeteksian serangan yang dapat terjadi kapan saja, sebagai contoh serangan ping flood. Ping flood sendiri dapat diartikan serangan penolakan terhadap layanan sederhana di mana penyerang membanjiri korban dengan paket "echo request" (ping) pada protocol ICMP. Untuk mengatasi permasalahan yang dihadapi oleh pihak pusat kendali Polres Kota Lubuklinggau dalam melakukan keamanan server CCTV tersebut, solusi yang ditawarkan adalah dengan membangun suatu Intrusion Detection Server (IDS). IDS sendiri dapat membaca paket-paket data yang masuk maupun yang keluar secara otomatis yang nantinya akan memberikan sebuah laporan (log) kepada administrator jaringan. Salah satu tools IDS yang banyak digunakan adalah Snort. Snort memiliki beberapa keunggulan dibandingkan software IDS yang lain antara lain source code yang berukuran kecil, kompatibel dengan banyak sistem operasi, cepat dalam mendeteksi serangan jaringan, mudah dikonfigurasi dan bersifat open source.

Kata kunci: *Keamanan Jaringan, IDS, Snort.*

PENERBIT

Jurusan Teknik Elektro-ITATS
Alamat:
Jl. Arief Rachman Hakim No.100,Surabaya 60117,
Telp/Fax: 031-5997244

Jurnal JREEC by Department of Elecreical Engineering is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

PENDAHULUAN

Keamanana jaringan merupakana hal yang sangat penting dalam dunia jaringan. Banyak faktor yang dapat mengganggu keamanan dan kestabilan dari suatu koneksi jaringan tersebut. Sistem keamanan jaringan yang baik dapat meminimalisis kerugian yang disebabkan oleh serangan keamanan jaringan[1]. Oleh karena itu, peran sistem keamanan jaringan komputer sebagai bagian dari sebuah sistem informasi merupakan bagian yang penting untuk menjaga validitas dan integritas data serta dapat menjamin ketersediaan layanan bagi penggunanya. Pusat Kendali (Command Center) Polres Lubuklinggau yang selanjutnya disebut Pusat Kendali adalah suatu sistem terpadu berbasis teknologi informasi yang terintegrasi untuk mendukung kegiatan operasional kepolisian dalam rangka pelayanan masyarakat. Salah satu pelayanan terpadu tersebut berupa unit pemantauan layanan CCTV. Seluruh pantauan dari kamera CCTV dilakukan penyimpanan ke dalam sebuah server[2]. Permasalahan yang timbul di pusat kendali CCTV ini adalah belum adanya pengamanan terhadap pendeteksian serangan yang dapat terjadi kapan saja, sebagai contoh serangan ping flood. Ping flood sendiri dapat diartikan serangan penolakan terhadap layanan sederhana di mana penyerang membanjiri korban dengan paket "echo request" (ping) pada protocol ICMP. Dengan tidak adanya pengamanan terhadap kemungkinan serangan yang terjadi, tentunya dapat merugikan pihak dari Polres Kota Lubuklinggau dimana data-data di dalam server tersebut mengalami kerusakan atau bahkan sampai hilang.

Untuk mengatasi permasalahan yang dihadapi oleh pihak pusat kendali Polres Kota Lubuklinggau dalam melakukan keamanan server CCTV tersebut, solusi yang ditawarkan adalah dengan membangun suatu Intrusion Detection Server (IDS). IDS sendiri dapat membaca paket-paket data yang masuk maupun yang keluar secara otomatis yang nantinya akan memberikan sebuah laporan (log) kepada administrator jaringan. Salah satu tools IDS yang banyak digunakan adalah Snort. Snort memiliki beberapa keunggulan dibandingkan software IDS yang lain antara lain source code yang berukuran kecil, kompatibel dengan banyak sistem operasi, cepat dalam mendeteksi serangan jaringan, mudah dikonfigurasi dan bersifat open source.

TINJAUAN PUSTAKA

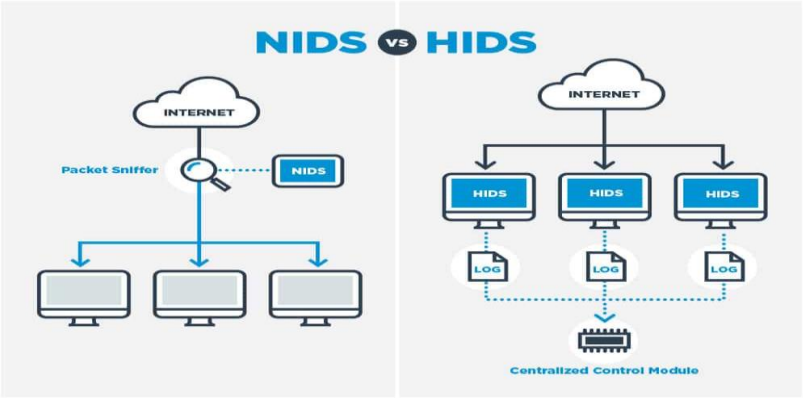
Intrusion Detection System (IDS)

Intrusion Detection System adalah perangkat lunak atau perangkat keras yang dirancang untuk mendeteksi aktifitas berbahaya baik dalam hal serangan terhadap suatu sistem maupun terhadap suatu jaringan komputer. IDS dapat melakukan inspeksi terhadap lalu lintas jaringan inbound dan outbound dalam suatu jaringan. Ketika menemukan serangan, maka akan memberikan peringatan apakah aktifitas tersebut termasuk berbahaya atau tidak berdasarkan beberapa level, yaitu low, medium, high, dan serious. IDS juga dapat didefinisikan sebagai tool, metode, sumber daya yang memberikan bantuan untuk melakukan identifikasi, memberikan laporan terhadap aktivitas jaringan komputer[3]. Aplikasiyang digunakan untuk melakukan penyerangan ke komputer server dalam penelitian ini adalah aplikasi Loic (Low Orbit Ion cannon). Loic (Low Orbit Ion)merupakan sebuah tool atau aplikasi yang berfungsi untuk melumpuhkan server sebuah situs website dengan mengirimkan packet sebanyak mungkin sesuai dengan kemauan si penyerang ke komputer server yang dituju melalui domain atau ip server komputer target. IDS dibagi menjadi 2 (dua) bagian yaitu *Network-based IDS* (NIDS) dan *Host-based IDS (HIDS)*

Network-based IDS (NIDS)

Berfungsi untuk memantau dan memonitor semua lalu lintas jaringan pada keseluruhan jaringan, NIDS akan menangkap semua lalu lintas jaringan dan mengirimkan *copy* dari lalu lintas yang ditangkap dan mengirimkan ke IDS. NIDS biasanya diletakkan di dalam segmen jaringan di

mana *server* berada atau di pintu masuk jaringan. Contoh IDS yaitu *Snort*. Gambar 2.1 menyajikan ilustrasi NIDS dan HIDS.



Gambar 2.1 Ilustrasi NIDS dan HIDS

Host-based IDS (HIDS)

Berfungsi memantau dan menganalisis lalu lintas jaringan yang masuk dan keluar dari *host*. Perbedaan utama HIDS dengan NIDS adalah NIDS memonitor seluruh segmen jaringan, sedangkan HIDS hanya memonitor pada *host* tertentu, biasanya diletakkan di *server-server* kritis di jaringan seperti *firewall* dan *web server*. HIDS juga menangkap lalu lintas jaringan seperti *snapshot* dan dibandingkan dengan *snapshot* sebelummnya, jika terdapat perbedaan maka akan mengirimkan alert kepada administrator.

Snort

Snort merupakan sebuah perangkat lunak yang berfungsi untuk mengamati aktifitas dalam suatu jaringan komputer. Komponen pada Snort terdiri dari beberapa bagian yaitu:

a. *Packet Decoder*
   Berfungsi untuk mengekstrak paket dari jaringan dalam bentuk file berformat 'tcpdump' dan mengirimkan paket ke *preprocessor*.
b. *Preprocessor*
   Berfungsi untuk memodifikasi paket yang rusak menggunakan beberapa operasi dan kemudian mengirimkan ulang ke *Detection Engine*.
c. *Detection Engine*
   Berfungsi untuk mendeteksi ancaman aktivitas yang ada dalam paket dengan menggunakan *snort rules*.
d. *Logging and Alerting System*
   Berfungsi untuk menghasilkan alarm atau log aktivitas intrusi yang terdeteksi oleh *Detection Engine*.
e. *Output Modules*
   Berfungsi untuk menyimpan output yang dihasilkan oleh *Logging and Alerting System*.
   Adapun dalam mengoperasikan Snort ada beberapa cara, antara lain:
   1) *Sniffer mode*: Pada mode ini, *Snort* akan menangkap semua paket pada jaringan tertentu.
   2) *Packet Logger Mode*: Pada mode ini, *Snort* akan menangkap semua paket yang melintas, dan menyimpan di *storage*.
   3) *Network Intrusion Detection Mode*: Pada mode ini, *Snort* akan menjalan *file* konfigurasi yang sudah diatur pada *file* 'snort.conf'.

Serangan Pada Jaringan Komputer

Ada berbagai macam serangan pada jaringan komputer, beberpa diantara di jelaskan sebagai berikut**:**

LAND Attack

Salah satu serangan terhadap suatu server yang terhubung dalam suatu jaringan untuk menghentikan layanan, sehingga terjadi gangguan terhadap layanan atau jaringan computer. Tipe serangan semacam ini disebut sebagai Denial of Service (DoS) attack[4]. LAND attack dikategorikan sebagai serangan SYN (SYN attack) karena menggunakan packet SYN (synchronization) pada waktu melakukan 3-way handshake untuk membentuk suatu hubungan berbasis TCP/IP.

Ping of Death

Ping of Death merupakan suatu serangan (Denial of Service) DoS yang memanfaatkan fitur yang ada di TCP/IP yaitu packet fragmentation atau pemecahan paket. Penyerang dapat mengirimkan berbagai paket ICMP (digunakan untuk melakukan ping) yang terfragmentasisehingga waktu paket-paket tersebut disatukan kembali, maka ukuran paket seluruhnya melebihi batas 65536 byte.

Teardrop

Teardrop attack adalah suatu serangan bertipe Denial of Service (DoS) terhadap suatu server/komputer yang memanfaatkan fitur yang ada di TCP/IP yaitu packet fragmentation atau pemecahan paket, dan kelemahan yang ada di TCP/IP pada waktu paket-paket yang terfragmentasi tersebut disatukan kembali[5]. Dalam suatu pengiriman data dari satu komputer ke komputer yang lain melalui jaringan berbasis TCP/IP, maka data tersebut akan dipecah-pecah menjadi beberapa paket yang lebih kecil di komputer asal, dan paket-paket tersebut dikirim dan kemudian disatukan kembali di komputer tujuan. Server bisa diproteksi dari tipe serangan teardrop ini dengan paket filtering melalui firewall yang sudah dikonfigurasi untuk memantau dan memblokir paket-paket yang berbahaya seperti ini.

Half-Open Connection

Dalam serangan half-open connection, penyerang mengirimkan ke server yang hendak diserang banyak paket SYN yang telah dispoof atau direkayasa sehingga alamat asal (source address) menjadi tidak valid[6]. Tipe serangan half-open connection atau SYN attack ini dapat dicegah dengan paket filtering dan firewall, sehingga paket-paket SYN yang invalid tersebut dapat diblokir oleh firewall sebelum membanjiri server.

UDP Bomb Attack

Untuk melakukan serangan UDP Bomb terhadap suatu server, seorang penyerang mengirim sebuah paket UDP (User Datagram Protocol) yang telah dispoof atau direkayasa sehingga berisikan nilai-nilai yang tidak valid di field-field tertentu[7]. Jika server yang tidak terproteksi masih menggunakan sistem operasi (operating system) lama yang tidak dapat menangani paketpaket UDP yang tidak valid ini, maka server akan langsung crash.

METODE

Metode Pengumpulan data

Untuk mendapatkan data yang akurat maka dalam penyusunan proposal skirpsi ini penulis mengunakan beberapa metode pengumpulan data diantaranya adalah sebagai berikit ini :

a. Observasi

   Merupakan teknik atau pendekatan untuk mendapatkan data primer dengan mengamati langsung objek datanya sehingga data dapat diproleh secara orisinil pada saat terjadinya dan mencatatkan hasil observasi tersebut. Dengan melakukan observasi langsung untuk mencari informasi data baik alat dan bahan serta segala sesuatu yang digunakan dalam penelitian ini.

b. Wawancara

   Wawancara digunakan sebagai teknik pengumpulan data pada penelitian ini. Selain itu penulis juga melakukan wawancara yang menyangkut masalah potensi serangan terhadap server CCTV di *command center* Polres Kota Lubuklinggau.

c. Studi Literatur
Menggunakan metode pengumpulan data Literatur yaitu dengan mencari referensi dari buku, majalah, jurnal, artikel, internet, dan sumber lainnya yang berkaitan dengan judul yang diambil, kemudian dirangkum untuk disusun dan di sempurnakan

Metode Pengembangan Sistem

Dalam penelitian ini penulis menggunakan metode *Live Forensic* untuk membangun sistem deteksi serangan server CCTV di command center Polres Lubuklinggau[8]. Berikut merupakan langkah-langkah pengembangan perangkat tersebut:

a. *Preparation and collection*
Pada tahapan ini dilakukan pengumpulan data, analisis sistem berjalan, dan analisis sistem yang akan dirancang.
b. *Examination*
Pada tahapan ini dilakukan inspeksi terhadap perangkat keras dan perangkat lunak yang digunakan.
c. *Analysis*
Pada tahap ini dilakukan analisis terhadap hasil skenario penyerangan terhadap sistem. Scenario dilakukan dengan melakukan serangan pada protocol ICMP yang ada di komputer server CCTV.
c. *Log*
Setelah dilakukan analisis, maka dilakukan pencatatan (log) terhadap hasil analisis dari deteksi serangan yang terjadi[9]. Pada tahapan ini akan diketahui apakah Snort yang digunakan untuk mendeteksi serangan ping flood efektif atau tidak.

Metode Pengujian Sistem

Pengujian dilakukan dengan menggunakan metode eksperimen yang dimaksudkan apakah sistem Snort dapat mendeteksi serangan *ping flood* dengan baik[10]. Pengujian dilakukan dengan menggunakan *ping flood* yang akan dilakukan oleh komputer *client*. Tabel 3.3 menyajikan perancangan pengujian sistem.

Tabel 3.3 Perancangan Pengujian

No	Komponen Pengujian	Proses Pengujian	Hasil Pengujian
1	Komputer server	Kinerja *Snort* pada IDS.	Snort dapat mendeteksi serangan yang dilakukan *client.*
2	Komputer Client	Melakukan *ping flood* ke komputer *server*.	Paket-paket data yang berisi serangan *ping flood*.

Rancangan Sistem

Rancangan sistem dimaksudkan untuk membangun eksperimen terhadap model serangan ping flood yang akan dideteksi[11]. Rancangan eksperimen ini dibagi menjadi 2 bagian, antara lain perancangan perangkat keras dan perancangan perangkat lunak. Dalam proses perancangan perangkat keras, eksperimen ini dibangun dengan menggunakan 2 PC / laptop. PC1 digunakan sebagai server yang diinstalasi Snort dan PC2 sebagai client yang bertindak sebagai attacker. Gambar 3.4 menyajikan rancangan perangkat keras yang akan dibangun, dapat dilihat dibawah ini :

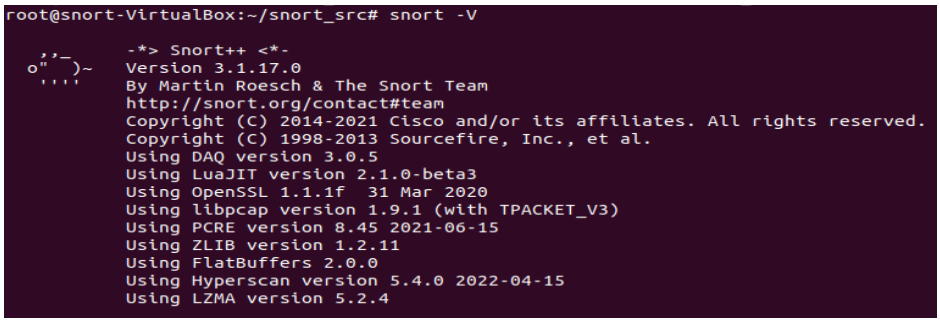


Gambar 3.4 Rancangan Topologi Eksperimen

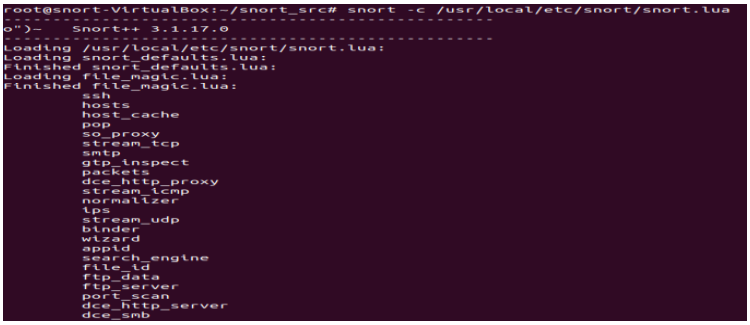
HASIL DAN PEMBAHASAN

Pembahasan Data I

Hasil Instalasi Snort, Untuk tahap pertama dari adalah hasil instalasi snort 3 dapat ditunjukkan pada gambar 4.1, dan pengujian snort berjalan dengan baik ditunjukan pada gambar 4.2.



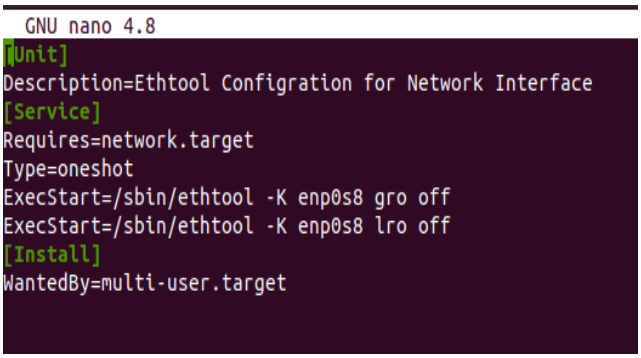
Gambar 4.1 Hasil Instalasi Snort



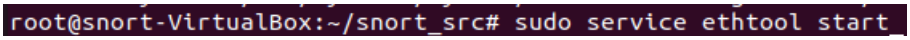
Gambar 4.2 Pengujian Hasil Instalasi

Gambar 4.3 dan gambar 4.4 menunjukkan konfigurasi perangkat ethernet yang akan di monitor atau diawasi oleh snort dan mengaktifkannya[12].

Pembahasan Data II



Gambar 4.3 Pengaturan Perangkat Ethernet



Gambar 4.4 Aktifasi Perangkat Ethernet

Gambar 4.5 menunjukkan direktori untuk menyimpan *rule* snort, *rule* ini digunakan untuk menyimpan perintah pada snort dalam mendeteksi serangan yang sudah didefinisikan.

```
sudo mkdir /usr/local/etc/rules
sudo mkdir /usr/local/etc/so_rules/
sudo mkdir /usr/local/etc/lists/

sudo touch /usr/local/etc/rules/local.rules
sudo touch /usr/local/etc/lists/default.blocklist

sudo mkdir /var/log/snort
```

Gambar 4.5 Direktori Pengaturan Snort

Gambar 4.6 adalah *rule* atau aturan yang penulis buat untuk mendeteksi serangan *ping flood,* serangan *ping flood* menggunakan paket data protokol ICMP yang dikirimkan secara *masive* sehingga server akan mengalami kelebihan paket data yang diterima sehingga menyebabkan pengguna lain yang mengakses server tersebut menjadi terhambat[13].

```
 GNU nano 4.8                                  /usr/local/etc/rules/local.rules
alert icmp any any -> any any ( msg:"Mendeteksi lalulintas Data ICMP  "; sid:10000001; metadata:policy security-ips alert; )
```

Gambar 4.6 Rule Snort Deteksi Ping Flood

Rule atau aturan yang penulis buat untuk mendeteksi *ping flood* adalah dengan mendeteksi paket data ICMP yang berlebihan, dan melewati ethernet yang sudah dimasukkan dalam pengaturan snort untuk diawasi[14].

Pembahasan Data III

Gambar 4.7 dan 4.8 menunjukkan penggunaan *rule* untuk mendeteksi serangan *ping flood* dan hasil deteksi serangan *ping flood* pada server[15].

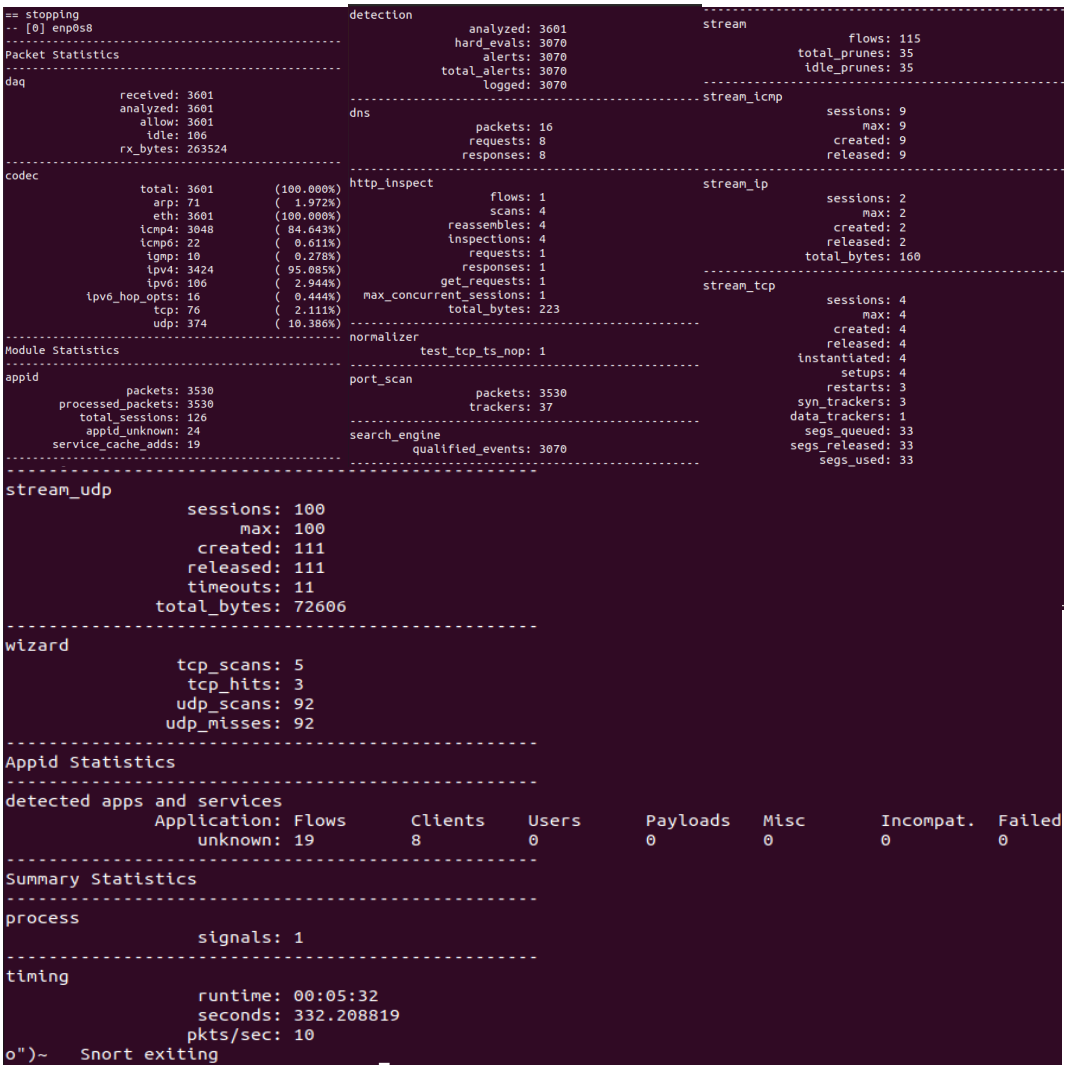
```
root@snort-VirtualBox:~/snort_src# sudo snort -c /usr/local/etc/snort/snort.lua -R /usr/local/etc/rules/local.rules \
> -i enp0s8 -A alert_fast -s 65535 -k none
```

Gambar 4.7 Penerapan *rule* snort untuk mendeteksi *ping flood* secara *real time*

```
04/15-09:16:12.238565 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.246 -> 192.168.88.248
04/15-09:16:12.238591 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.248 -> 192.168.88.246
04/15-09:16:12.270914 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.246 -> 192.168.88.248
04/15-09:16:12.270935 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.248 -> 192.168.88.246
04/15-09:16:12.303216 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.246 -> 192.168.88.248
04/15-09:16:12.303236 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.248 -> 192.168.88.246
04/15-09:16:12.335161 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.246 -> 192.168.88.248
04/15-09:16:12.335181 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.248 -> 192.168.88.246
04/15-09:16:12.367275 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.246 -> 192.168.88.248
04/15-09:16:12.367298 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.248 -> 192.168.88.246
04/15-09:16:12.398650 [**] [1:10000001:0] "Mendeteksi lalulintas Data ICMP  " [**] [Priority: 0] {ICMP} 192.168.88.246 -> 192.168.88.248
```

Gambar 4. 8 Hasil deteksi serangan *ping flood* secara *real time*

Gambar 4.9 menunjukkan statistik hasil deteksi *ping flood* secara *real time* menggunakan snort. Dari gambar 4.9 tersebut dapat dilihat ip address dari computer penyerang yaitu 192.168.88.246 dengan serangan ping flood yang membanjiri lalu lintas data di protocol ICMP. Dampak dari serangan ini akan meningkatnya penggunaan resource sumber daya computer server seperti CPU hingga 100 % bahkan overload[16].



Gambar 4.9 Hasil Deteksi Serangan Ping Flood

Dari pengujian yang telah dilakukan maka dapat ditarik kesimpulan sistem deteksi serangan ping flood dapat mendeteksi serangan tersebut secara real time[17].

KESIMPULAN

Hasil dari penelitian ini berguna Untuk mengatasi permasalahan yang dihadapi oleh pihak pusat kendali Polres Kota Lubuklinggau dalam melakukan keamanan server CCTV tersebut, selain itu juga untuk membuat sistem keamanan jaringan dalam mengatasi serangan ping flood di server CCTV Polres Kota Lubuklinggau dapat menggunakan Snort pada IDS. Solusi yang ditawarkan adalah dengan membangun suatu *Intrusion Detection Server* (IDS). IDS sendiri dapat membaca paket-paket data yang masuk maupun yang keluar secara otomatis yang nantinya akan memberikan sebuah laporan (log) kepada administrator jaringan. Salah satu tools IDS yang banyak digunakan adalah *Snort*. *Snort* memiliki beberapa keunggulan dibandingkan software IDS yang lain antara lain *source code* yang berukuran kecil, kompatibel dengan banyak sistem operasi, cepat dalam mendeteksi serangan jaringan, mudah dikonfigurasi dan bersifat *open source*.

DAFTAR PUSTAKA

[1] A. H. Hambali and S. Nurmiati, "Implementasi Intrusion Detection System (IDS) Pada Keamanan PC Server Terhadap Serangan Flooding Data," *Sainstech J. Penelit. dan Pengkaj. Sains dan Teknol.*, vol. 28, no. 1, pp. 35–43, 2018, doi: 10.37277/stch.v28i1.267.

[2] P. Panggabean, "Analisis Network Security Snort Metode Intrusion Detection System Untuk Optimasi Keamanan Jaringan Komputer," *Jursima*, vol. 6, no. 1, p. 1, 2018, doi: 10.47024/js.v6i1.107.

[3] S. M. Othman, F. Mutaher Ba-Alwi, N. T. Alsohybe, and A. T. Zahary, "Survey on Intrusion Detection System Types," *Int. J. Cyber-Security Digit. Forensics*, vol. 7, no. 4, pp. 444–462, 2018, [Online]. Available: https://www.researchgate.net/publication/329363322

[4] B. Wijaya and A. Pratama, "Deteksi Penyusupan Pada Server Menggunakan Metode Intrusion Detection System (Ids) Berbasis Snort," *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 9, no. 1, pp. 97–101, 2020, doi: 10.32736/sisfokom.v9i1.770.

[5] I. P. A. E. Pratama and N. K. M. Handayani, "Implementasi IDS Menggunakan Snort Pada Sistem Operasi Ubuntu," *J. Mantik Penusa*, vol. 3, no. 1, pp. 176–181, 2019.

[6] S. Khadafi, Y. D. Pratiwi, and E. Alfianto, "Keamanan Ftp Server Berbasiskan Ids Dan Ips Menggunakan Sistem Operasi Linux Ubuntu," *Netw. Eng. Res. Oper.*, vol. 6, no. 1, p. 11, 2021, doi: 10.21107/nero.v6i1.190.

[7] I. G. N. W. Arsa, "Arsitektur Konsolidasi Server dengan Virtualisasi untuk Penyedia Layanan Infrastruktur Cloud," *J. Sist. dan Inform.*, vol. 14, no. 1, pp. 35–40, 2019, doi: 10.30864/jsi.v14i1.240.

[8] O. A. Astra and Y. Mardiana, "Rancang Bangun dan Analisa Pengendali CCTV Berbasis Arduino Menggunakan Smartphone Android," *J. Media Infotama*, vol. 14, no. 1, 2018, doi: 10.37676/jmi.v14i1.470.

[9] D. V. Sandi and M. Arrofiq, "Implementasi Analisis NIDS Berbasis Snort Dengan Metode Fuzy Untuk Mengatasi Serangan LoRaWAN," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 2, no. 3, pp. 685–696, 2018, doi: 10.29207/resti.v2i3.504.

[10] F. Antony and R. Gustriansyah, "Deteksi Serangan Denial of Service pada Internet of Things Menggunakan Finite-State Automata," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 1, pp. 43–52, 2021, doi: 10.30812/matrik.v21i1.1078.

[11] I. Zuhriyanto, A. Yudhana, and I. Riadi, "Perancangan Digital Forensik pada Aplikasi Twitter Menggunakan Metode Live Forensics," *Semin. Nas. Inform. 2008 (semnasIF 2008)*, vol. 2018, no. November, pp. 86–91, 2018.

[12] D. Santoso, A. Noertjahyana, and J. Andjarwirawan, "Implementasi dan Analisa Snort dan Suricata Sebagai IDS dan IPS Untuk Mencegah Serangan DOS dan DDOS," *J. Infra*, vol. 10, no. 1, pp. 1–6, 2022, [Online]. Available: https://publication.petra.ac.id/index.php/teknik-informatika/article/view/12033

[13] I. Riadi, S. Sunardi, and M. E. Rauli, "Identifikasi Bukti Digital WhatsApp pada Sistem Operasi Proprietary Menggunakan Live Forensics," *J. Tek. Elektro*, vol. 10, no. 1, pp. 18–22, 2018, doi: 10.15294/jte.v10i1.14070.

[14] R. Suwanto, I. Ruslianto, and M. Diponegoro, "Implementasi Intrusion Prevention System (IPS) Menggunakan Snort Dan IPTable Pada Monitoring Jaringan Lokal Berbasis Website," *J. Komput. dan Apl.*, vol. 07, no. 1, pp. 97–107, 2019.

[15] I. K. K. A. Marta, I. N. B. Hartawan, and I. K. S. Satwika, "Analisis Sistem Monitoring Keamanan Server Dengan Sms Alert Berbasis Snort," *Inser. Inf. Syst. Emerg. Technol. J.*, vol. 1, no. 1, p. 25, 2020, doi: 10.23887/insert.v1i1.25874.

[16] W. W. Purba and R. Efendi, "Perancangan dan analisis sistem keamanan jaringan komputer menggunakan SNORT," *Aiti*, vol. 17, no. 2, pp. 143–158, 2021, doi: 10.24246/aiti.v17i2.143-158.

[17] Soni, Y. Prayudi, and B. Sugiantoro, "Teknik Akuisisi Virtualisasi Server Menggunakan Metode Live Forensic," *Teknomatika*, vol. 9, no. 2, 2017.